# Is a Picture Worth a Thousand Words: Visual Emotional Analysis using Transfer Learning and CNNs

Noah Jacobson, Katherine Kowalski, Hasna Rtabi
noahj08@stanford.edu, kasia4@stanford.edu, hasna@stanford.edu

## Motivation

- Historically, computers have been incapable of assessing human emotion, and emotion was thought to be outside the capabilities of a machine
- Modern advances in deep learning, however, show that this might not be the case.
- We built different models that label the emotions present in a piece of visual art and compare their performance
- Challenges:
  - complexities found in both art and emotion.
  - combine computer vision and NLP
  - diversity of images

## Background

There is string psychological evidence that images tend to elicit certain emotions in their viewer based on the style and content of the image. As social media networks become more widespread, there are more uses cases for neural networks that analyze emotions based on image data. Note that these images do not just include facial images – they include all types of images imaginable.

Now, in the paper, "Building a Large Scale Dataset for Image Emotion Recognition: The Fine Print and The Benchmark", authors describe a dataset of non-facial images that are labeled by their emotions. When the researches trained convolutional neural networks to identify the emotions in these images, the accuracies they found are shown in the chart above. These accuracies are not as high as they ideally would be, so researchers search for a better way to analyze the emotions in images. In our project, we worked with a Stanford researcher to explore one novel idea.

| Algorithms | Correct Samples | Accuracy |
|---|---|---|
| ImageNet-CNN | 1120/3490 | 32.1% |
| Noisy-Fine-tuned-CNN | 1600/3490 | 45.8% |
| Fine-tuned-CNN | 2034/3490 | 58.3% |

[8]

Results as published by You, Q., Luo, J., Jin, H. and Yang, J. on the dataset we will be using

### Our Project

Our project will take three approaches towards solving this problem

1. Residual neural network
2. Fractal Neural Network
3. Feature embedding and multi-class classification

## Data and Features

[8]

| Train 90% | Dev 10% | Test 10% |
|---|---|---|
| 67,937 | 8,491 | 8,492 |

Image (32,32,3)
Label = 0 amusement
Label = 1 angry
Label = 2 awe
Label = 3 contentment
Label = 4 disgust
Label = 5 excitement
Label = 6 fear
Label = 7 sadness

- Dataset of 84,922 labelled images, scaled down to 32x32x3 for computational efficiency and memory limitations [8]
- Each label indicates the most prevalent emotion in the images with an integer, for a total of 8 classes
  - 0: amusement, 1: angry, 2:awe, 3: contentment, 4: disgust, 5: excitement, 6: fear, 7: sadness

## Methods

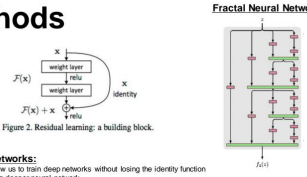



Figure 2. Residual learning: a building block.

**Residual Neural Networks:**

- "Skip" connections allow us to train deep networks without losing the identity function
- This allows us to train a deeper neural network
- It also helps us preserve textural information from previous layers which might be especially important for emotion

**Fractal Neural Networks**



**Feature embedding & Multi-Class Classification**

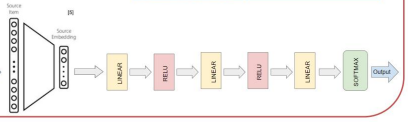
- Used a model pretrained on the MSCOCO dataset
- Mapped image vectors to a feature vector using a CNN
- Trained feature vectors with a three layer tensorflow neural network

Epochs: 1000
Mini Batch Size: 64
Learning Rate: .0001

- M: number of classes, in our case 8
- log: natural log
- $y_{i,c}$ 1 if class label 'c' is correct given observation 'o', else 0
- $p_{i,c}$ probability 'o' is of class 'c'

Cross entropy loss:

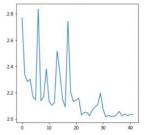
$$J = -\sum_{i=1}^{M} y_{i,c} log(p_{i,c})$$



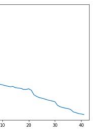

[8]

## Results

**Residual Neural Network**

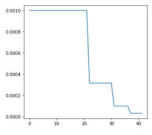


Epoch Number vs Validation Loss



Epoch Number vs Train Loss

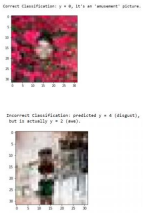
**Fractal Neural Network**

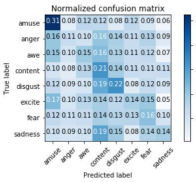


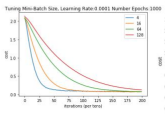
Epoch Number vs Learning Rate

| Algorithm | Train Accuracy | Dev Accuracy | Test Accuracy |
|---|---|---|---|
| Residual Neural Network | 52% | 25% | 24% |
| Fractal Neural Network | 13% | 13% | 13% |
| Feature Embedding & Multi-Class Classification | 46% | 20% | 18% |

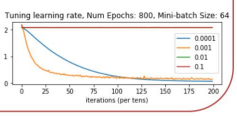
**Feature Embedding & Multi-Class Classification**




Optimal
Learning Rate: 0.001
Number of Epochs: 1000
Mini-Batch Size: 64




## Discussion

The purpose of this project was to determine whether it would be effective to caption images using image captioning neural networks and pass those captions through another neural net to predict the emotions in an image. In order to determine this, we created three different models. The first was a residual CNN with softmax output to predict which emotion was expressed in the image and the second used image embedding matrices from image captioning repositories. We found that we were not able to get the residual CNN model to surpass those made by the researchers who attempted this problem before us, so we moved on to other approaches.

We also tried to use fractal neural networks, which use fractal network layers in order to process an image. Fractal nets have a lot of the same benefits of resnets, however in our case we found fractal networks to be too deep to train. We found that training was extremely slow, and in the few tests we were able to run, we ran into an exploding gradient problem. As a result, we stopped pursuing this path.

The feature embedding followed by a neural net method was not as effective as we had hoped it to be. It seems that the algorithm is doing a little better than "guessing" what the emotional response is. This may be due to the fact that an emotional response to an image is very subjective and finding the key features that cause an image to be labeled "angry" or "content" is difficult. In addition, the feature embedding matrix was intended to be used to create captions for images – no to classify images based on emotion. As a result, it is likely that we lost relevant information and/or gained unnecessary information in the feature embedding. We found that the algorithm works best on emotional responses of amusement, contentment, and disgust. Furthermore, we see misclassification between similar categories such as fear and anger.

As a whole, there are so many different ways in which an image can express an emotion that our model was not able to learn all of them. Instead, we noticed that it was much more common for our model to simply memorize the training set despite increases in regularization, dropout, and data augmentation. For the future of this project, we would recommend getting a larger dataset or narrowing down the potential types of images that we will try to identify.

## Future Work

When we decided to use a residual convolutional neural network for this project, we had assumed that a resnet would be the best option for analyzing emotions in images due to the depth of resnets and the ability of resnets to perform identity transforms in its layers. We found, however, that the resnets did not train as well as we would have hoped. Consequently, on the convolutional neural networks side, it would be important for future work to explore different network architectures. In addition, an important factor that we did not consider in doing this study is that some images express more than one emotion. As a result, it is possible that our model was accurate but that is was told it was inaccurate because it labeled one emotion in an image which expressed several emotions. Thus, for future work, it will be important to allow images to have multiple labels as to not falsely penalize correct weights.

As for the feature embedding and multi-class classification model, some possible ways to improve performance include modifying the loss function to penalize predicting a positive emotion when the true emotion is positive and vice versa, as well as shortening the list of emotions by, for example, combining similar emotions such as anger and fear. Increasing the resolution of the images from 32x32x3 could also help make the images more recognizable and thus improve performance in the transfer learning model (resizing to 64x64x3 did not improve performance whatsoever in the CNN, so we decided not to try resizing for the transfer model). Finally, implement dropout and regularization would help reduce the problem of overfitting to the training set.

## References

[1] He, Kaiming, et al. "Identity Mappings in Deep Residual Netw orks." *[Astro-Ph/0505112] A Determination of the Hubble Constant from Cepheid Distances and a Model of the Local Peculiar Velocity Field*, American Physical Society, 25 July 2016, arxiv.org/abs/1603.05027.

[2] Jay, Prakash. "Image Classification Architectures Review – Prakash Jay – Medium." *Medium.com*, Medium, 19 July 2018, medium.com/@14prakash/image-classification-architectures-review-d8b65075998f.

[3] Kroner, Thomas. "Fractal Neural Netw orks." *Introduction to Fractal Geometry*, www.fractal.org/Life-Science-Technology-Publications/Fractal-Neural-Networks.htm.

[4] Lin, T., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. and Dollár, P. (2018). *Microsoft COCO: Common Objects in Context* [online] Arxiv.org. Available at: https://arxiv.org/abs/1405.0312 [Accessed 12 Dec. 2018].

[5] Mackey, C. (2018). *Building an image caption generator with Deep Learning in Tensorflow* [online] freeCodeCamp.org. Available at: https://medium.freecodecamp.org/building-an-image-caption-generator-with-deep-learning-in-tensorflow-a142722e9b11 [Accessed 12 Dec. 2018].

[6] Raghakot. "Raghakot/Keras-Resnet." *GitHub*, 6 July 2017, github.com/raghakot/keras-resnet.

[7] Snf. "Snf/Keras-Fractalnet." *GitHub*, 17 Sept. 2017, github.com/snf/keras-fractalnet.

[8] You, Q., Luo, J., Jin, H. and Yang, J. (2016). *Building a Large Scale Dataset for Image Emotion Recognition: The Fine Print and The Benchmark*. [online] Arxiv.org. Available at: https://arxiv.org/abs/1605.02677 [Accessed 12 Dec. 2018].

[9] Zhang, Xiangyu, et al. "Deep Residual Learning for Image Recognition." *[Astro-Ph/0505112] A Determination of the Hubble Constant from Cepheid Distances and a Model of the Local Peculiar Velocity Field*, American Physical Society, 10 Dec. 2015, arxiv.org/abs/1512.03385.

## Acknowledgements

We would like to thank Vinay K. Chaudhri for the sponsored project, for supplying the data, as well as the idea. We would also like to thank our project TA, Pedro Garzon for the guidance. Lastly, we would like to thank the CS230 staff and teaching assistance for a quarter of learning!